

FILEID**STRCHESTA

F 16

SSSSSSSS	TTTTTTTT	RRRRRRRR	CCCCCCCC	HH	HH	EEEEEEEEE	SSSSSSSS	TTTTTTTT	AAAAAA
SSSSSSSS	TTTTTTTT	RRRRRRRR	CCCCCCCC	HH	HH	EEEEEEEEE	SSSSSSSS	TTTTTTTT	AAAAAA
SS	TT	RR	RR	CC	HH	EE	SS	TT	AA
SS	TT	RR	RR	CC	HH	EE	SS	TT	AA
SS	TT	RR	RR	CC	HH	EE	SS	TT	AA
SS	TT	RR	RR	CC	HH	EE	SS	TT	AA
SSSSSS	TT	RRRRRRRR	CC	HHHHHHHHHH	HH	EEEEEEE	SSSSSS	TT	AA
SSSSSS	TT	RRRRRRRR	CC	HHHHHHHHHH	HH	EEEEEEE	SSSSSS	TT	AA
SS	TT	RR	RR	CC	HH	EE	SS	TT	AAAAAA
SS	TT	RR	RR	CC	HH	EE	SS	TT	AAAAAA
SS	TT	RR	RR	CC	HH	EE	SS	TT	AA
SS	TT	RR	RR	CC	HH	EE	SS	TT	AA
SSSSSSSS	TT	RR	RR	CCCCCCCC	HH	HH	EEEEEEEEE	SSSSSSSS	TT
SSSSSSSS	TT	RR	RR	CCCCCCCC	HH	HH	EEEEEEEEE	SSSSSSSS	TT

LL	IIIIII	SSSSSSSS
LL	IIIIII	SSSSSSSS
LL	II	SS
LLLLLLLL	IIIIII	SSSSSSSS
LLLLLLLL	IIIIII	SSSSSSSS

```
0001 0 XTITLE 'STR$CHECK STATUS -Check status and convert to STR$ status'  
0002 0 MODULE STR$CHECK_STATUS ( ! Check status and convert to STR$ status  
0003 0  
0004 0 IDENT = '1-003' ! File: STRCHESTA.B32 Edit: RKR1003  
0005 0  
0006 0 ) =  
0007 1 BEGIN  
0008 1  
0009 1 *****  
0010 1 *  
0011 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
0012 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
0013 1 * ALL RIGHTS RESERVED.  
0014 1 *  
0015 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
0016 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
0017 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
0018 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
0019 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
0020 1 * TRANSFERRED.  
0021 1 *  
0022 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
0023 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
0024 1 * CORPORATION.  
0025 1 *  
0026 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
0027 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
0028 1 *  
0029 1 *  
0030 1 *****  
0031 1 .  
0032 1  
0033 1 ++  
0034 1 FACILITY: String support library  
0035 1  
0036 1 ABSTRACT:  
0037 1  
0038 1 This module contains a single routine which inspects the  
0039 1 incoming status ( usually a LIB$ or SSS status) and converts  
0040 1 it into the appropriate STR$ status. If it is one of the  
0041 1 fatal LIB$ statuses, the corresponding STR$ status gets  
0042 1 signalled.  
0043 1 Unrecognized statuses are left alone.  
0044 1  
0045 1 ENVIRONMENT: Runs at any access mode - AST reentrant  
0046 1  
0047 1 AUTHOR: R. Reichert, CREATION DATE: 3-MAY-1981  
0048 1  
0049 1 MODIFIED BY:  
0050 1  
0051 1 1-001 - Original. RKR 3-MAY-1981  
0052 1 1-002 - Revise what error statuses get signaled. RKR 3-SEP-1981.  
0053 1 1-003 - Add explicit external declaration for LIB$STOP.  
0054 1 RKR 4-NOV-1981.  
0055 1  
0056 1 --  
0057 1
```

```
59      0058 1 %SBTTL 'Declarations'  
60      0059 1  
61      0060 1 SWITCHES:  
62      0061 1  
63      0062 1  
64      0063 1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL,  
65      0064 1      NONEXTERNAL = WORD_RELATIVE);  
66      0065 1  
67      0066 1 LINKAGES:  
68      0067 1  
69      0068 1 REQUIRE 'RTLIN:STRLNK';      ! The linkage declarations  
70      0253 1  
71      0254 1 TABLE OF CONTENTS:  
72      0255 1  
73      0256 1  
74      0257 1 FORWARD ROUTINE  
75      0258 1      STR$CHECK_STATUS_R2 : STR$CHECK_STATUS_LINKAGE;  
76      0259 1  
77      0260 1  
78      0261 1 INCLUDE FILES:  
79      0262 1  
80      0263 1  
81      0264 1 LIBRARY 'RTLSTARLE';      ! System symbols, typically from  
82      0265 1      SYSSLIBRARY:STARLET.L32  
83      0266 1 REQUIRE 'RTLIN:RTLPSECT';      ! Define PSECT declarations macros  
84      0361 1  
85      0362 1 MACROS:  
86      0363 1  
87      0364 1      NONE  
88      0365 1  
89      0366 1 EQUATED SYMBOLS:  
90      0367 1  
91      0368 1      NONE  
92      0369 1  
93      0370 1 FIELDS:  
94      0371 1  
95      0372 1      NONE  
96      0373 1  
97      0374 1 PSECTS:  
98      0375 1  
99      0376 1 DECLARE PSECTS (STR);      ! Declare psects for STR$ facility  
100     0377 1 OWN STORAGE:  
101     0378 1  
102     0379 1      NONE  
103     0380 1  
104     0381 1 EXTERNAL REFERENCES:  
105     0382 1  
106     0383 1  
107     0384 1 EXTERNAL ROUTINE  
108     0385 1      LIB$STOP;      ! to signal errors  
109     0386 1  
110     0387 1 EXTERNAL LITERAL  
111     0388 1      LIB$_INSVIRMEM,      ! Condition value symbols  
112     0389 1      LIB$_INVSTRDES,      ! Insufficient virtual memory  
113     0390 1      LIB$_INVARG,      ! Invalid string descriptor  
114     0391 1      LIB$_STRTRU,      ! Invalid argument  
115     0392 1      LIB$_WRONUMARG,      ! String truncated  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1097  
1098  
1099  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1197  
1198  
1199  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1279  
1280  
1281  
12
```

STR\$CHECK_STAT STR\$CHECK_STATUS -Check status and convert to Declarations 1-003 16-Sep-1984 01:31:12 14-Sep-1984 12:40:01 I 16 VAX-11 Bliss-32 V4.0-742 [LIBRTL.SRC]STRCHESTA.B32;1 Page 3 (2)

116	0393	1	STR\$NORMAL	Success
117	0394	1	STR\$INSVIRMEM,	Insufficient virtual memory
118	0395	1	STR\$ILLSTRCLA,	Illegal string class
119	0396	1	STR\$TRU,	Truncation
120	0397	1	STR\$WRONUMARG;	Wrong number of arguments
121	0398	1		

```

123      0399 1 %SBTTL 'STR$CHECK_STATUS_R2 - Check Status'
124      0400 1 GLOBAL ROUTINE STR$CHECK_STATUS_R2 ( ! Check status
125      0401 1
126      0402 1     IN_STATUS
127      0403 1
128      0404 1
129      0405 1
130      0406 1 !++
131      0407 1     FUNCTIONAL DESCRIPTION:
132      0408 1
133      0409 1     This routine inspects the incoming status. If it is one of
134      0410 1     a recognized set of fatal LIB$ statuses, the corresponding
135      0411 1     STR$ status is signalled. Some LIB$ statuses are converted
136      0412 1     to the corresponding STR$ status. If the status is unknown,
137      0413 1     it is simply returned.
138      0414 1
139      0415 1     CALLING SEQUENCE:
140      0416 1
141      0417 1     STR$CHECK_STATUS_R2 ( IN_STATUS.rlc.v )
142      0418 1
143      0419 1     FORMAL PARAMETERS:
144      0420 1
145      0421 1     IN_STATUS.rlc.v      The input status
146      0422 1
147      0423 1     IMPLICIT INPUTS:
148      0424 1
149      0425 1     NONE
150      0426 1
151      0427 1     IMPLICIT OUTPUTS:
152      0428 1
153      0429 1
154      0430 1
155      0431 1     COMPLETION STATUS:
156      0432 1
157      0433 1     OUT_STATUS.wlc.v      The output status
158      0434 1
159      0435 1
160      0436 1     SIDE EFFECTS:
161      0437 1
162      0438 1     If it finds IN_STATUS to be LIB$_INSVIRMEM, LIB$_INVSTRDES,
163      0439 1     LIB$_WRONUMARG or LIB$_INVARG it signals the corresponding STR$  
condition code.
164      0440 1
165      0441 1
166      0442 1     --
167      0443 1
168      0444 2     BEGIN
169      0445 2
170      0446 2     LOCAL
171      0447 2     OUT_STATUS ;      ! the status returned
172      0448 2
173      0449 3     OUT_STATUS = (SELECTONE .IN_STATUS OF
174      0450 3     SET
175      0451 3     [SS$_NORMAL,
176      0452 3     STR$_NORMAL]: .IN_STATUS ; ! most common first
177      0453 3
178      0454 3     [LIB$_STRTRU]: STR$_TRU ; ! next most common
179      0455 3

```

STR\$CHECK_STAT STR\$CHECK_STATUS -Check status and convert to K 16
 1-003 STR\$CHECK_STATUS_R2 - Check Status 16-Sep-1984 01:31:12 VAX-11 Bliss-32 V4.0-742
 [LIBRTL.SRC]STRCHESTA.B32;1

Page 5
 (3)

```
:
 180      0456 3      [LIB$_INSVIRMEM]: LIB$STOP (STR$_INSVIRMEM) ; !least
 181      0457 3      [LIB$_INVSTRDES]: LIB$STOP (STR$_ILLSTRCLA) ; ! common
 182      0458 3      [LIB$_INVARG]: LIB$STOP (STR$_ILLSTRCLA) ;
 183      0459 3      [LIB$_WRONUMARG]: LIB$STOP (STR$_WRONUMARG) ;
 184      0460 3
 185      0461 3      [01]HERWISE]: .IN_STATUS; ! unlikely
 186      0462 2      TES):
 187      0463 2      RETURN OUT_STATUS;
 188      0464 1      END: ! End of routine STR$CHECK_STATUS_R2
```

```
:
.TITLE STR$CHECK_STATUS STR$CHECK_STATUS -Check stat
us and convert to
.IDENT \1-003\

.EXTRN LIB$STOP, LIB$INSVIRMEM
.EXTRN LIB$INVSTRDES, LIB$INVARG
.EXTRN LIB$STRTRU, LIB$WRONUMARG
.EXTRN STR$NORMAL, STR$INSVIRMEM
.EXTRN STR$ILLSTRCLA, STR$TRU
.EXTRN STR$WRONUMARG

.PSECT _STR$CODE,NOWRT, SHR, PIC,2
```

52	50	DO 00000 STR\$CHECK STATUS R2::		
01	52	D1 00003	MOVL R0, R2	: 0400
00000000G 8F	52	D1 00006	CMPL IN_STATUS, #1	: 0451
00000000G 8F	53	13 00008	BEQL 6\$	
00000000G 8F	52	D1 0000F	CMPL IN_STATUS, #STR\$NORMAL	: 0454
	08	12 00011	BEQL 6\$	
	50 00000000G	8F	CMPL IN_STATUS, #LIB\$STRTRU	
	00000000G	D0 0001A	BNEQ 1\$	
		05 00021	MOVL #STR\$TRU, OUT_STATUS	
		RSB		
00000000G 8F	52	D1 00022	1\$: CMPL IN_STATUS, #LIB\$INSVIRMEM	: 0456
	08	12 00029	BNEQ 2\$	
	00000000G	8F	PUSHL #STR\$INSVIRMEM	
00000000G 8F	29	11 00031	BRB 5\$	
00000000G 8F	52	D1 00033	CMPL IN_STATUS, #LIB\$INVSTRDES	: 0457
00000000G 8F	09	13 0003A	BEQL 3\$	
	08	D1 0003C	CMPL IN_STATUS, #LIB\$INVARG	: 0458
	00000000G	8F	BNEQ 4\$	
		DD 00043	PUSHL #STR\$ILLSTRCLA	
00000000G 8F	0F	11 00048	BRB 5\$	
	00000000G	8F	CMPL IN_STATUS, #LIB\$WRONUMARG	: 0459
		DD 00045	BNEQ 6\$	
00000000G 00	01	DD 00056	PUSHL #STR\$WRONUMARG	
	00000000G	01	CALLS #1, LIB\$STOP	
		FB 0005C	RSB	
		05 00063	MOVL IN_STATUS, OUT_STATUS	: 0461
		50	RSB	: 0464

: Routine Size: 104 bytes, Routine Base: _STR\$CODE + 0000

```
:
 189      0465 1 END      ! End of module STR$CHECK_STATUS
 190      0466 1
```

STR\$CHECK_STAT STR\$CHECK_STATUS -{check status and convert to L 16
1-003 STR\$CHECK_STATUS_R2 - Check Status 16-Sep-1984 01:31:12 VAX-11 Bliss-32 V4.0-742
: 191 0467 0 ELUDOM [LIBRTL.SRC]STRCHESTA.B32;1

Page 6
(3)

PSECT SUMMARY

Name	Bytes	Attributes
_STR\$CODE	104	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	-----	Symbols	-----	Pages	Processing
	Total	Loaded	Percent	Mapped	Time
\$_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	1	0	581	00:00.8

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LISS:STRCHESTA/OBJ=OBJ\$:STRCHESTA MSRC\$:STRCHESTA/UPDATE=(ENH\$:STRCHESTA)

Size: 104 code + 0 data bytes
Run Time: 00:03.4
Elapsed Time: 00:16.5
Lines/CPU Min: 8339
Lexemes/CPU-Min: 25500
Memory Used: 50 pages
Compilation Complete

0213 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

